样例程序：ADC 模块使用

1.简介

这个文档的目的是帮助初上手的兄弟熟悉下DAVE-Tasking-memtool的流程，内容浅显，图文并茂，高手绕路

本程序实现功能如下：

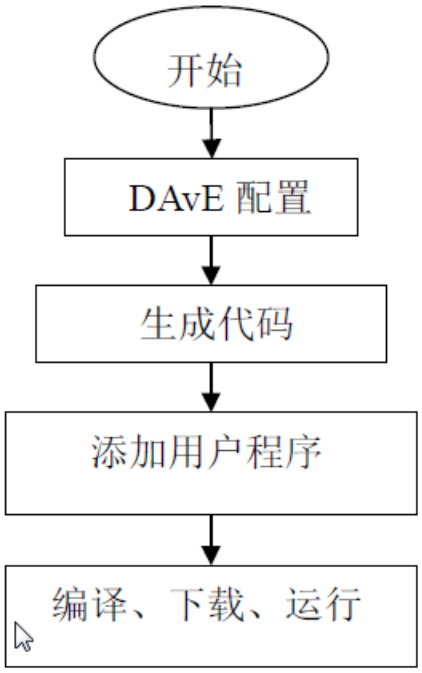
使用ADC0，请求源0软件产生转换请求，读出当前电位计的电压值并以此作为延迟时间。旋转电位计可以明显看到LED 灯以不同频率闪烁。

使用到的模块：AD C 和 IO

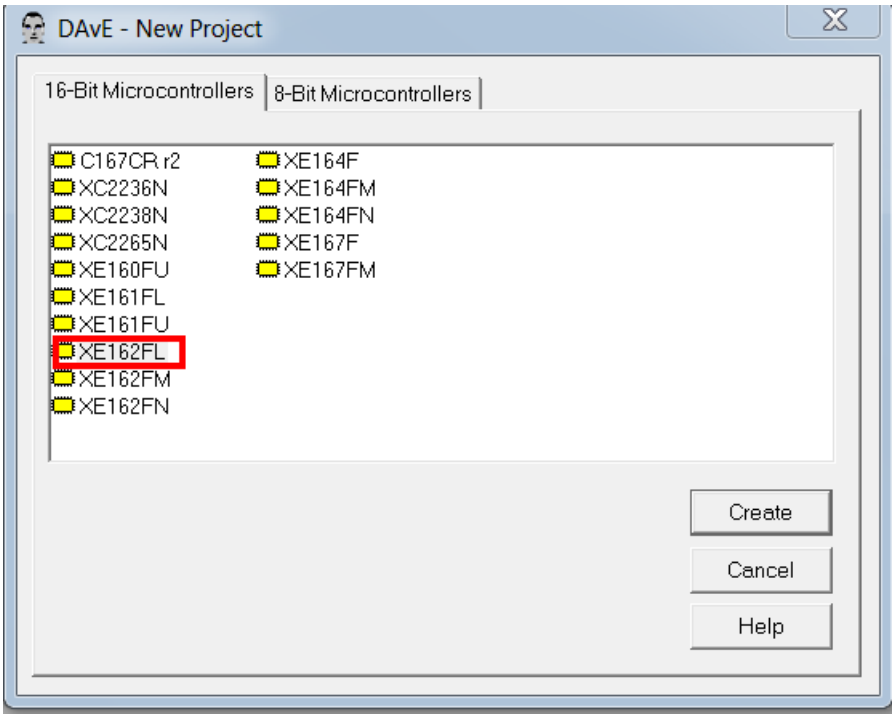
2 ADC 内核功能的特性如下所述：

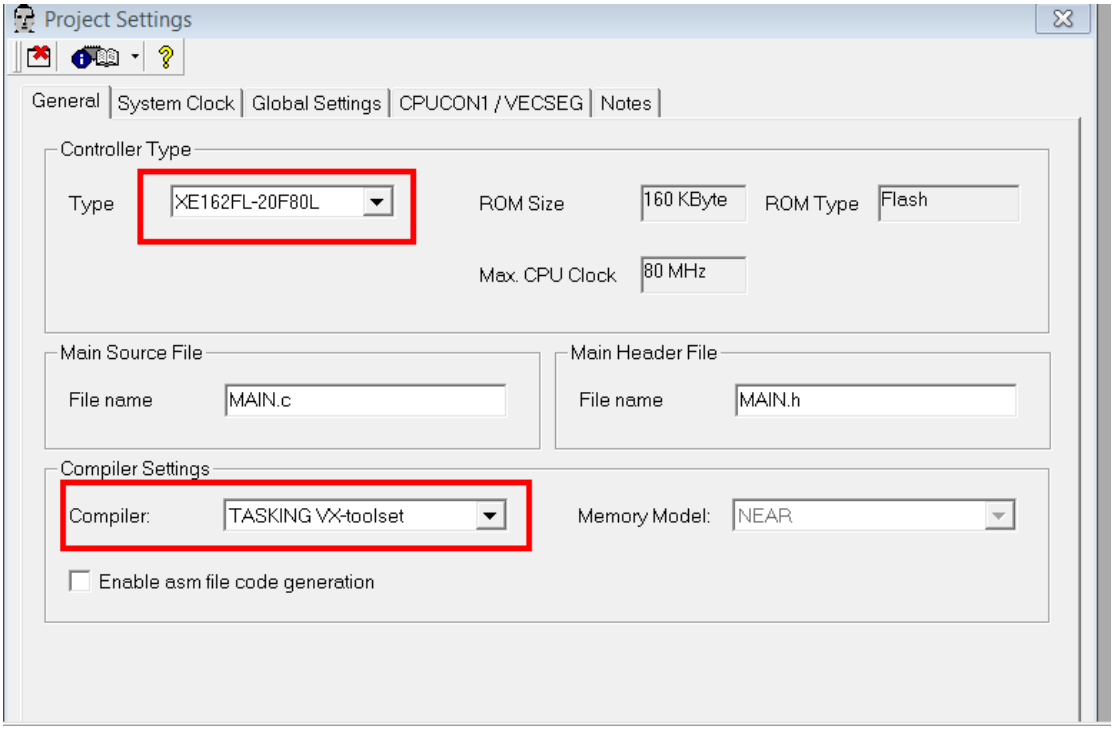
- 输入电压的范围可从 0V 到模拟供电电压（V_{DDPA}=3.0V 至 5.0V）
- 对于每个通道来说，标准（V_{AREF}）和备用（CH0）参考电压源均可选，以支持比例测量和不同的信号尺度
- 多达 19 个模拟输入通道（指 162FL，160U 8 个）
- 外部模拟复用器控制，包括可调整的采样时间和扫描支持。
- 转换速度和采样时间均可调整，以适应不同的传感器和参考源
- 转换时间低于 1 μs（由结果带宽和采样时间决定）
- 灵活的多源选择和仲裁
  - 单通道转换（单个或重复的）
  - 可配置的自动扫描转换（单个或重复的）
  - 可编程设置的仲裁转换序列（单个或重复的）
  - 由软件、定时器事件或外部事件触发的转换
  - 用于最大吞吐量的等待启动模式或用于降低转换延迟的取消插入重启模式
- 强大的结果处理
  - 可选择的结果带宽（8/10/12）
  - 8 个独立结果寄存器，并可组合创建结果 FIFO
  - 对于可编程设置的边界值执行可配置的极限检查
  - 通过添加可选择数量的转换结果，以降低数据速率
- 基于可选择事件的灵活的中断产生（PEC 支持）
- 内建的安全特性
  - 通过可编程设置的默认级别检测断线
  - 复用器测试模式可验证信号通路完整性
- 挂起和省电模式的支持

3.操作流程

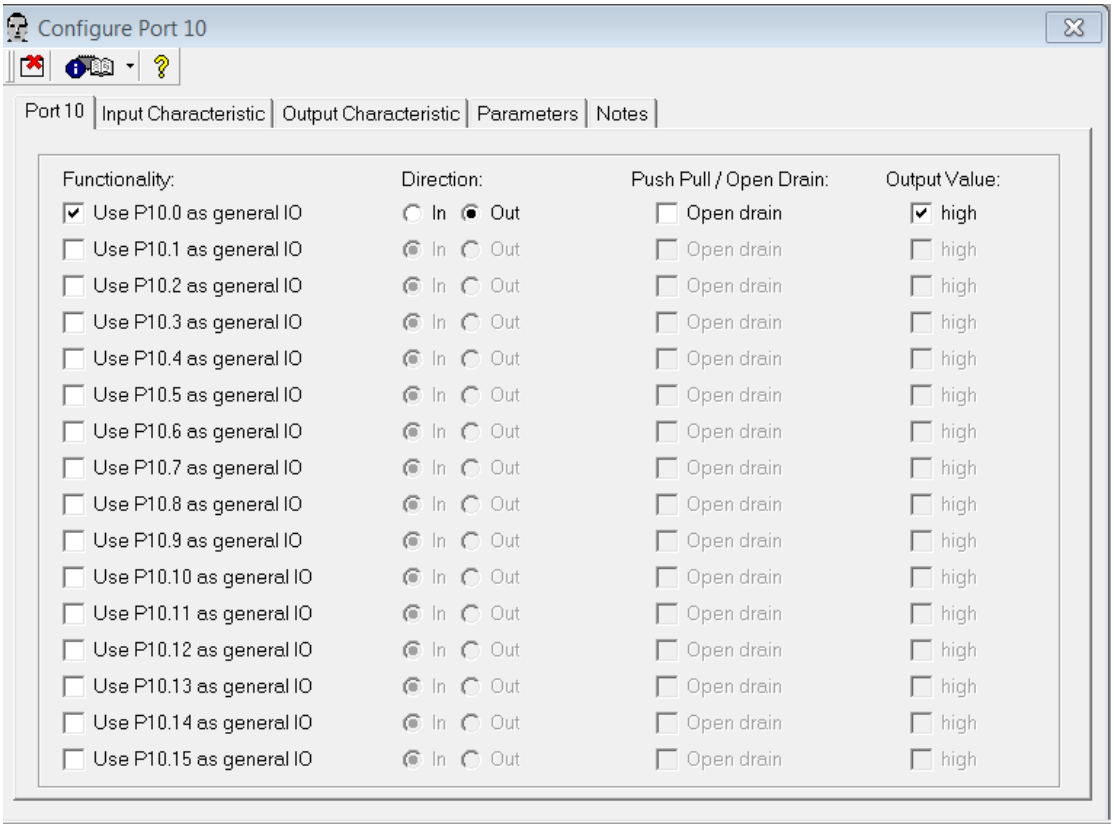


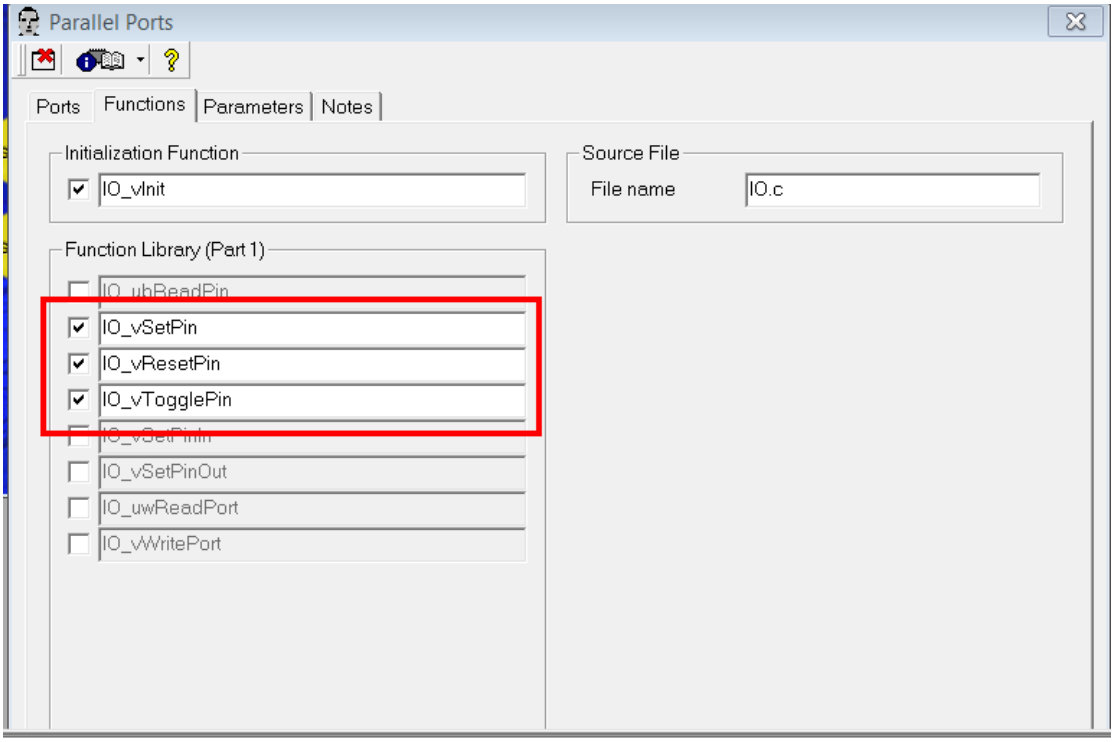
4.DAVE 配置



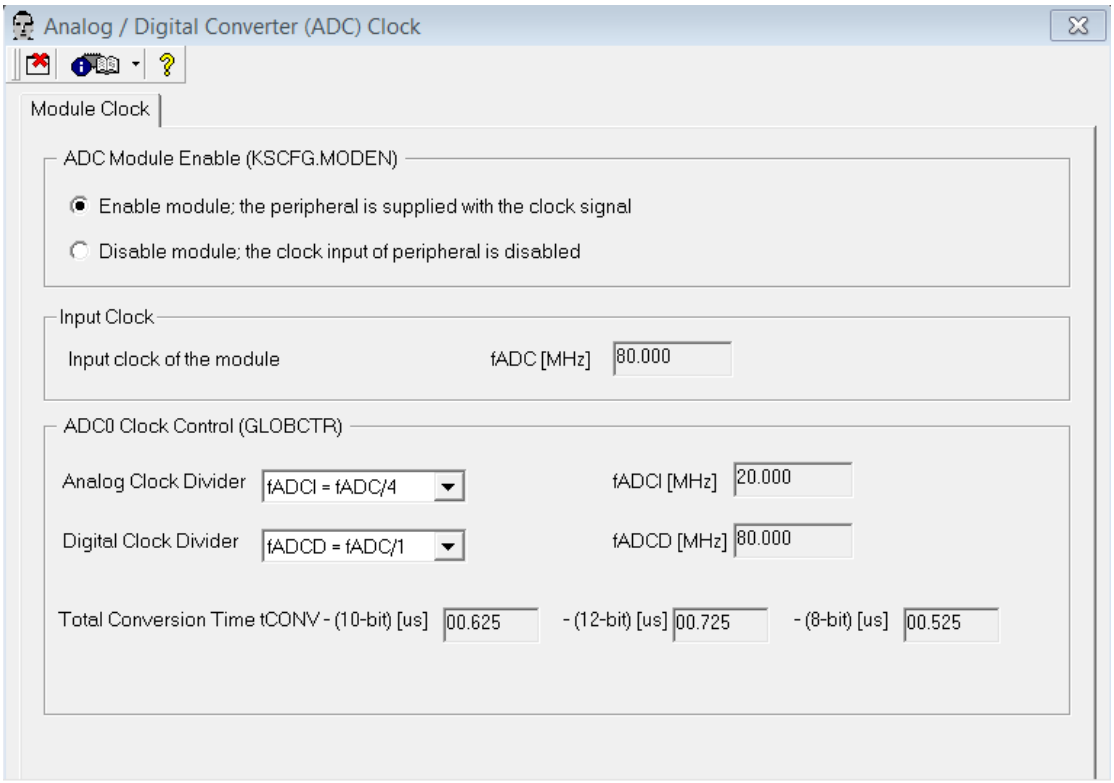

Project Settings
General
System Clock
Global Settings
CPUCON1 / VECSEG
Notes
Controller Type
Type
XE162FL-20F80L
ROM Size
160 KByte
ROM Type
Flash
Max. CPU Clock
80 MHz
Main Source File
File name
MAIN.c
Main Header File
File name
MAIN.h
Compiler Settings
Compiler:
TASKING VX-toolset
Memory Model:
NEAR
Enable asm file code generation

设置 I0 口先

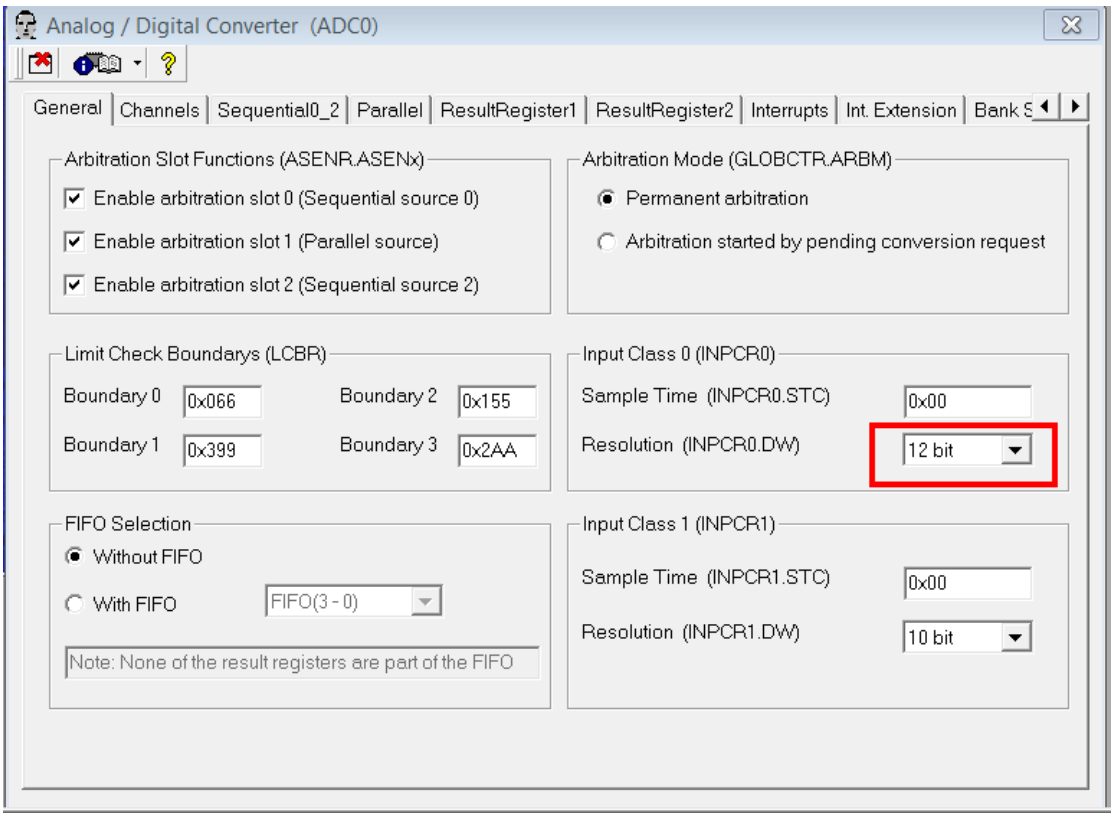

Configure Port 10
Port 10
Input Characteristic
Output Characteristic
Parameters
Notes
Functionality:
Direction:
Push Pull / Open Drain:
Output Value:
Use P10.0 as general IO
In
Out
Open drain
high
Use P10.1 as general IO
Use P10.2 as general IO
Use P10.3 as general IO
Use P10.4 as general IO
Use P10.5 as general IO
Use P10.6 as general IO
Use P10.7 as general IO
Use P10.8 as general IO
Use P10.9 as general IO
Use P10.10 as general IO
Use P10.11 as general IO
Use P10.12 as general IO
Use P10.13 as general IO
Use P10.14 as general IO
Use P10.15 as general IO



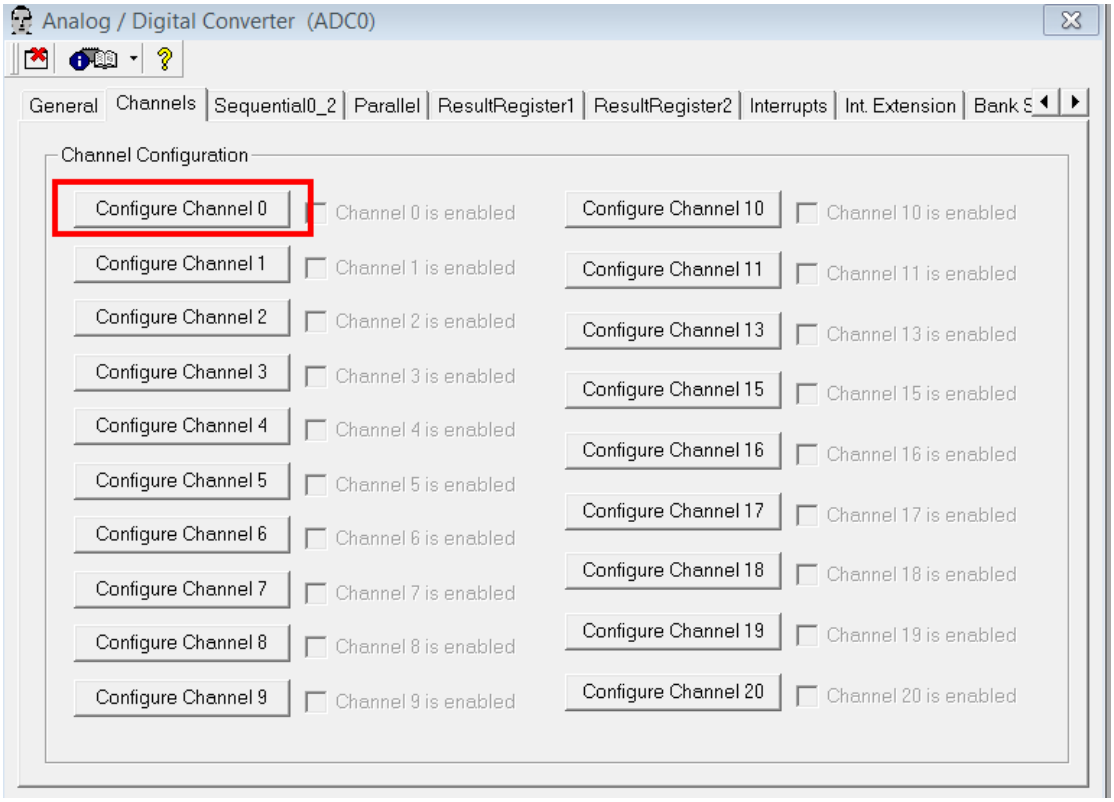
点开 ADC clock

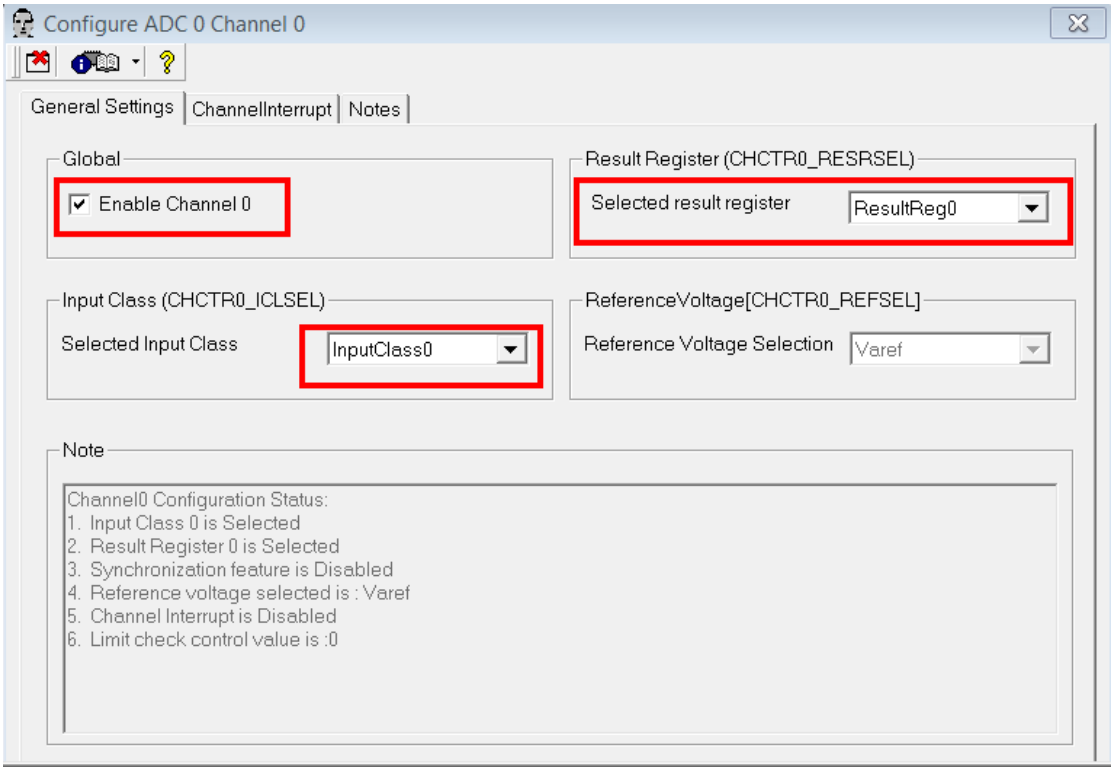


点击 ADC0

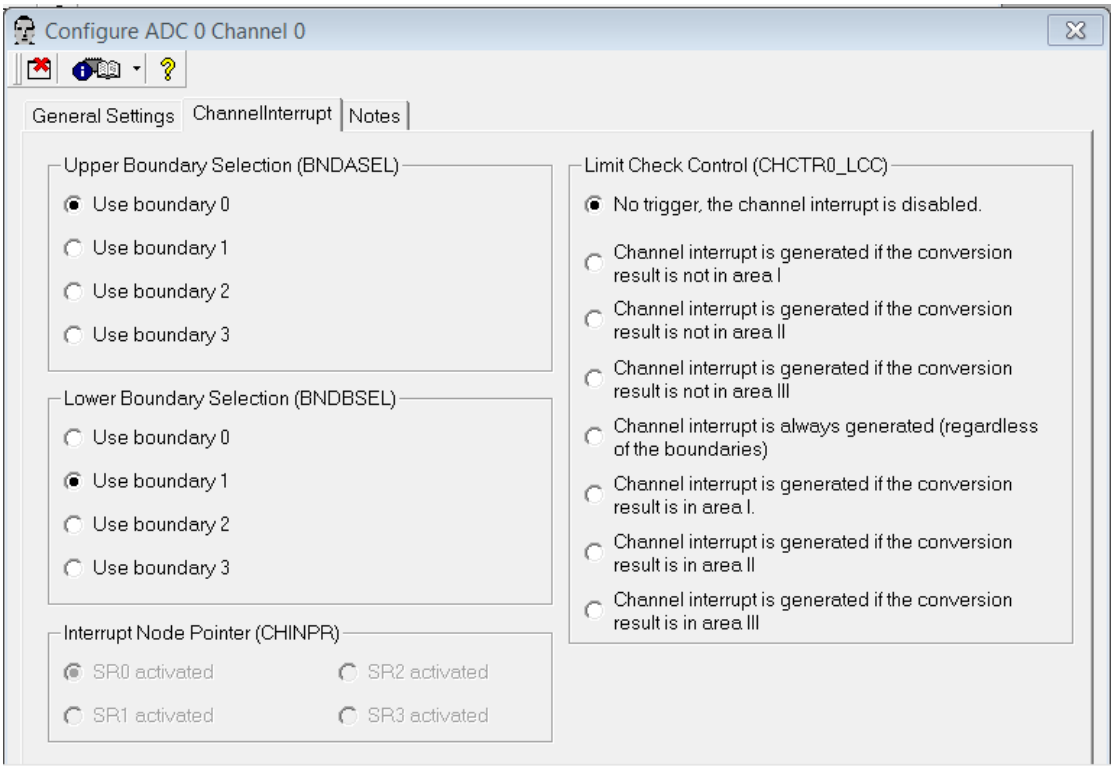

Analog / Digital Converter (ADC0)
General
Channels
Sequential0_2
Parallel
ResultRegister1
ResultRegister2
Interrupts
Int. Extension
Bank S
Arbitration Slot Functions (ASENR.ASENx)
Enable arbitration slot 0 (Sequential source 0)
Enable arbitration slot 1 (Parallel source)
Enable arbitration slot 2 (Sequential source 2)
Arbitration Mode (GLOBCTR.ARBM)
Permanent arbitration
Arbitration started by pending conversion request
Limit Check Boundarys (LCBR)
Boundary 0
0x066
Boundary 1
0x399
Boundary 2
0x155
Boundary 3
0x2AA
Input Class 0 (INPCR0)
Sample Time (INPCR0.STC)
0x00
Resolution (INPCR0.DW)
12 bit
FIFO Selection
Without FIFO
With FIFO
FIFO(3 - 0)
Note: None of the result registers are part of the FIFO
Input Class 1 (INPCR1)
Sample Time (INPCR1.STC)
0x00
Resolution (INPCR1.DW)
10 bit

点击 Channe1

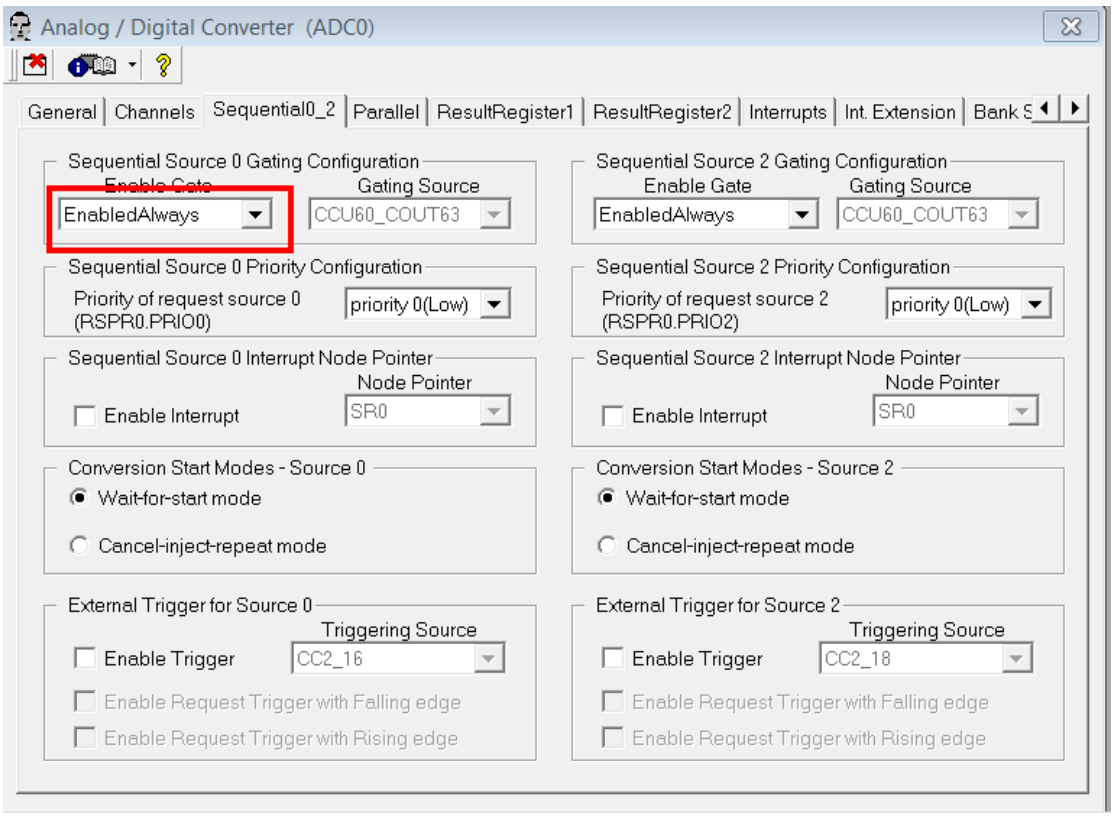

Analog / Digital Converter (ADC0)
General
Channels
Sequential0_2
Parallel
ResultRegister1
ResultRegister2
Interrupts
Int. Extension
Bank S
Channel Configuration
Configure Channel 0
Channel 0 is enabled
Configure Channel 1
Channel 1 is enabled
Configure Channel 2
Channel 2 is enabled
Configure Channel 3
Channel 3 is enabled
Configure Channel 4
Channel 4 is enabled
Configure Channel 5
Channel 5 is enabled
Configure Channel 6
Channel 6 is enabled
Configure Channel 7
Channel 7 is enabled
Configure Channel 8
Channel 8 is enabled
Configure Channel 9
Channel 9 is enabled
Configure Channel 10
Channel 10 is enabled
Configure Channel 11
Channel 11 is enabled
Configure Channel 13
Channel 13 is enabled
Configure Channel 15
Channel 15 is enabled
Configure Channel 16
Channel 16 is enabled
Configure Channel 17
Channel 17 is enabled
Configure Channel 18
Channel 18 is enabled
Configure Channel 19
Channel 19 is enabled
Configure Channel 20
Channel 20 is enabled


Configure ADC 0 Channel 0
General Settings
ChannelInterrupt
Notes
Global
Enable Channel 0
Result Register (CHCTR0_RESRSEL)
Selected result register
ResultReg0
Input Class (CHCTR0_ICLSEL)
Selected Input Class
InputClass0
ReferenceVoltage[CHCTR0_REFSEL]
Reference Voltage Selection
Varef
Note
Channel0 Configuration Status:
1. Input Class 0 is Selected
2. Result Register 0 is Selected
3. Synchronization feature is Disabled
4. Reference voltage selected is : Varef
5. Channel Interrupt is Disabled
6. Limit check control value is :0

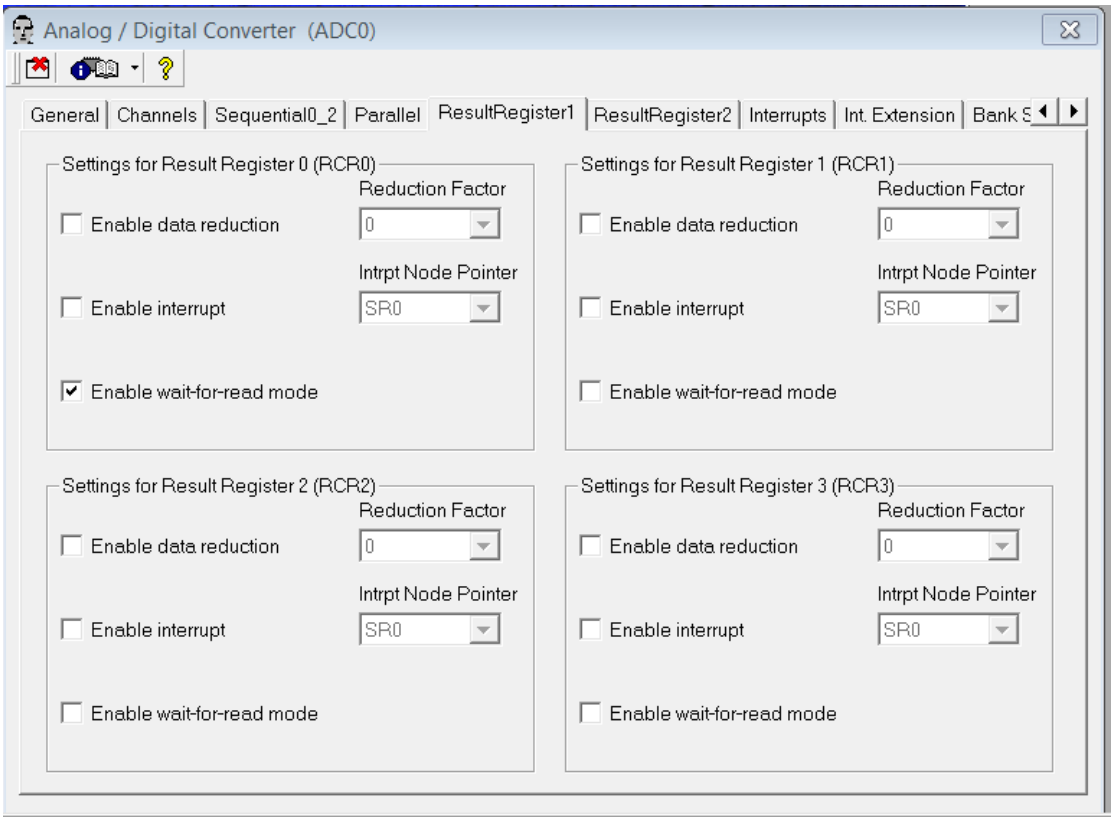
不设边界


Configure ADC 0 Channel 0
General Settings
ChannelInterrupt
Notes
Upper Boundary Selection (BNDASEL)
Use boundary 0
Use boundary 1
Use boundary 2
Use boundary 3
Lower Boundary Selection (BNDBSEL)
Use boundary 0
Use boundary 1
Use boundary 2
Use boundary 3
Interrupt Node Pointer (CHINPR)
SR0 activated
SR1 activated
SR2 activated
SR3 activated
Limit Check Control (CHCTR0_LCC)
No trigger, the channel interrupt is disabled.
Channel interrupt is generated if the conversion result is not in area I
Channel interrupt is generated if the conversion result is not in area II
Channel interrupt is generated if the conversion result is not in area III
Channel interrupt is always generated (regardless of the boundaries)
Channel interrupt is generated if the conversion result is in area I.
Channel interrupt is generated if the conversion result is in area II
Channel interrupt is generated if the conversion result is in area III

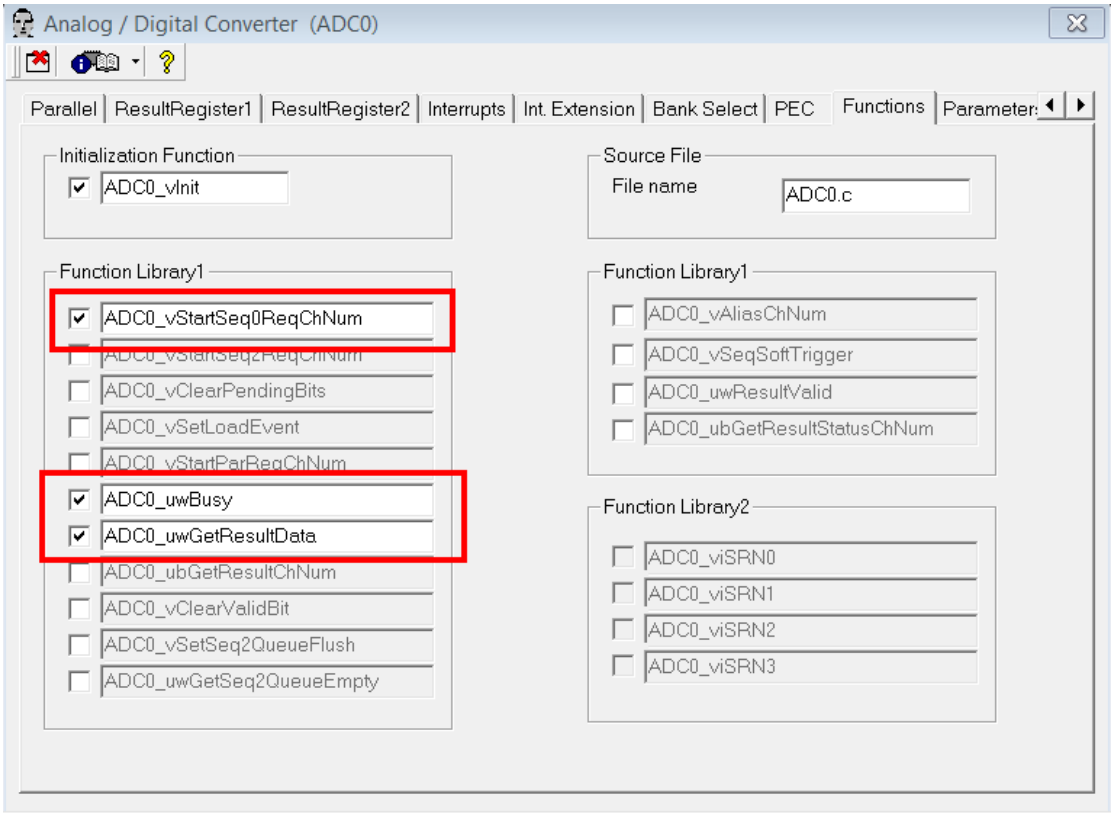
请求源 0 配置



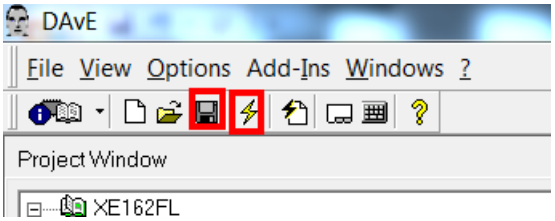
结果寄存器配置



选择需要 DAVE 生成的函数

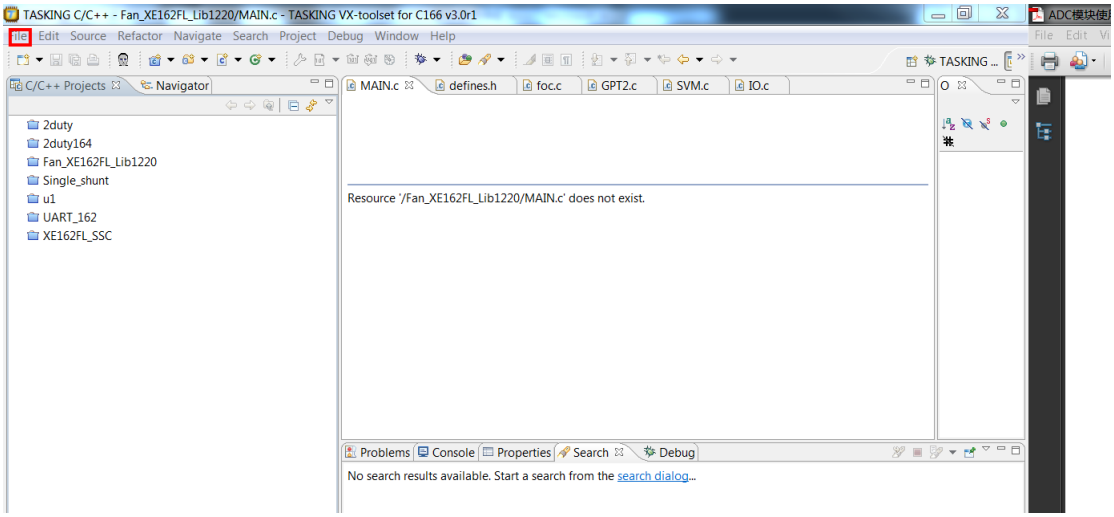


5.利用 DAVE 生成代码

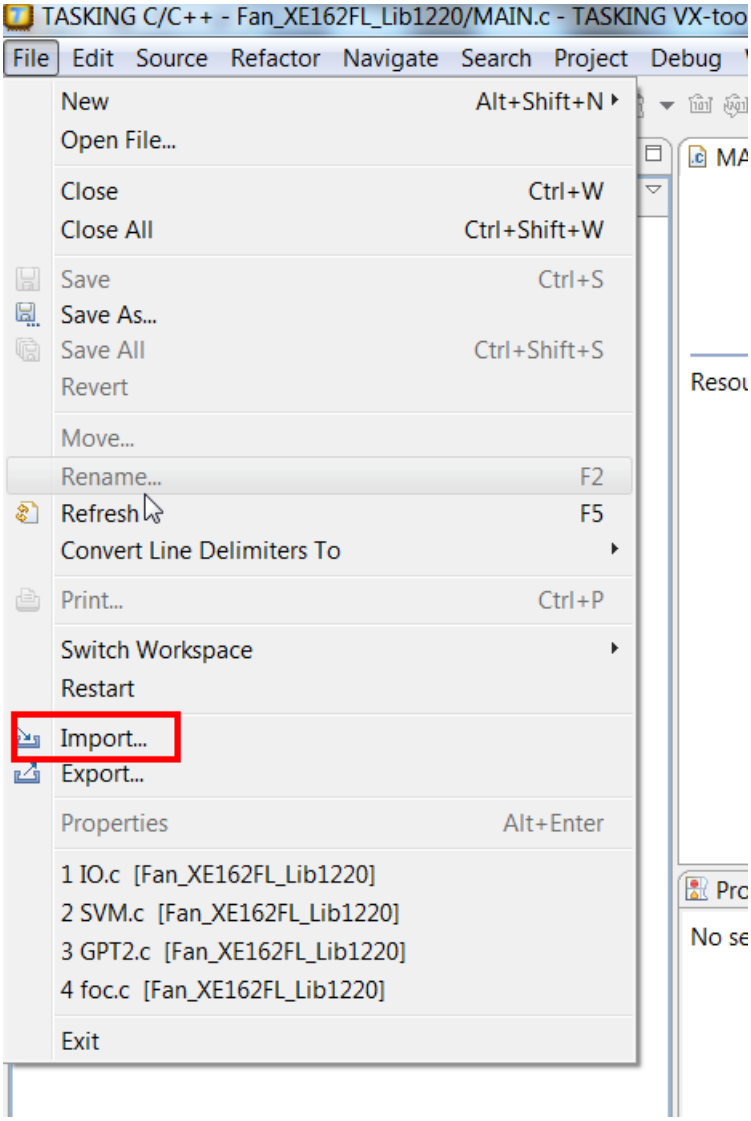


点击 左图标保存project，点击右图标DAVE 自动生成代码，生成的代码即包括前面所选择的函数。

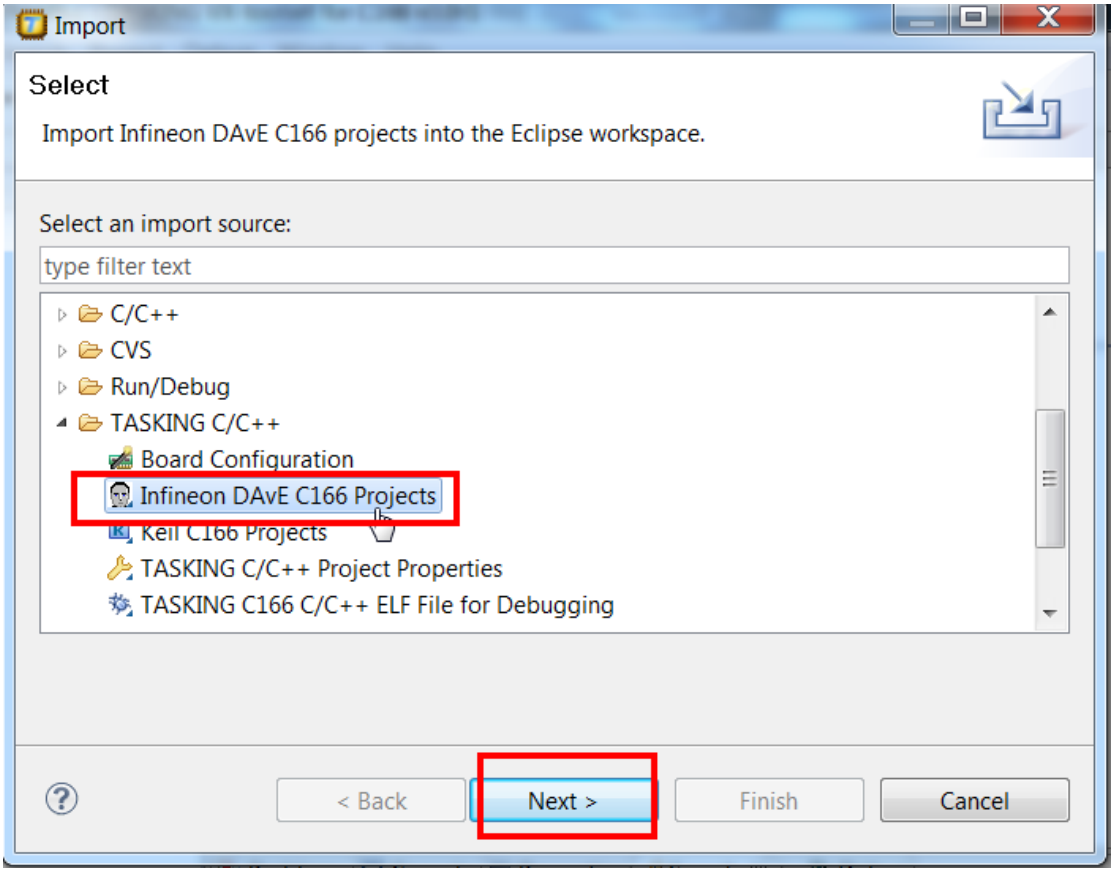
6 自动代码生成后，打开 TASKING VX-tool

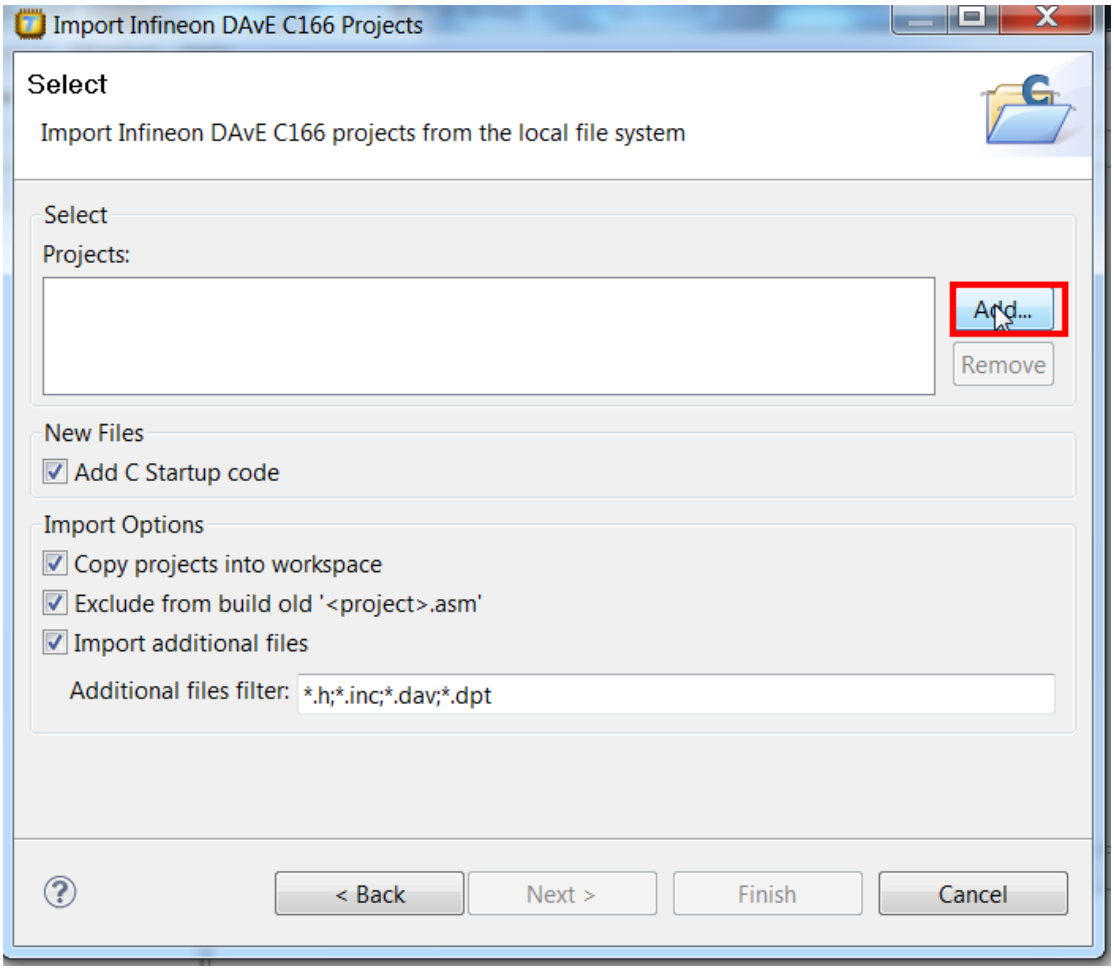


打开 File

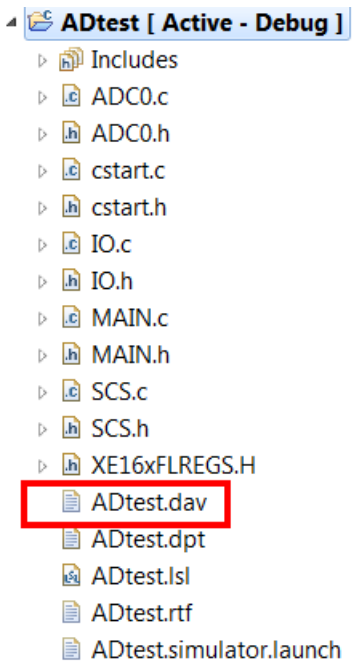


指定刚才生成的 DAVE 文件

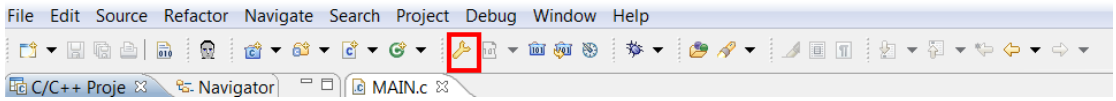

Import
Select
Import Infineon DAvE C166 projects into the Eclipse workspace.
Select an import source:
type filter text
C/C++
CVS
Run/Debug
TASKING C/C++
Board Configuration
Infineon DAvE C166 Projects
Keil C166 Projects
TASKING C/C++ Project Properties
TASKING C166 C/C++ ELF File for Debugging
< Back
Next >
Finish
Cancel

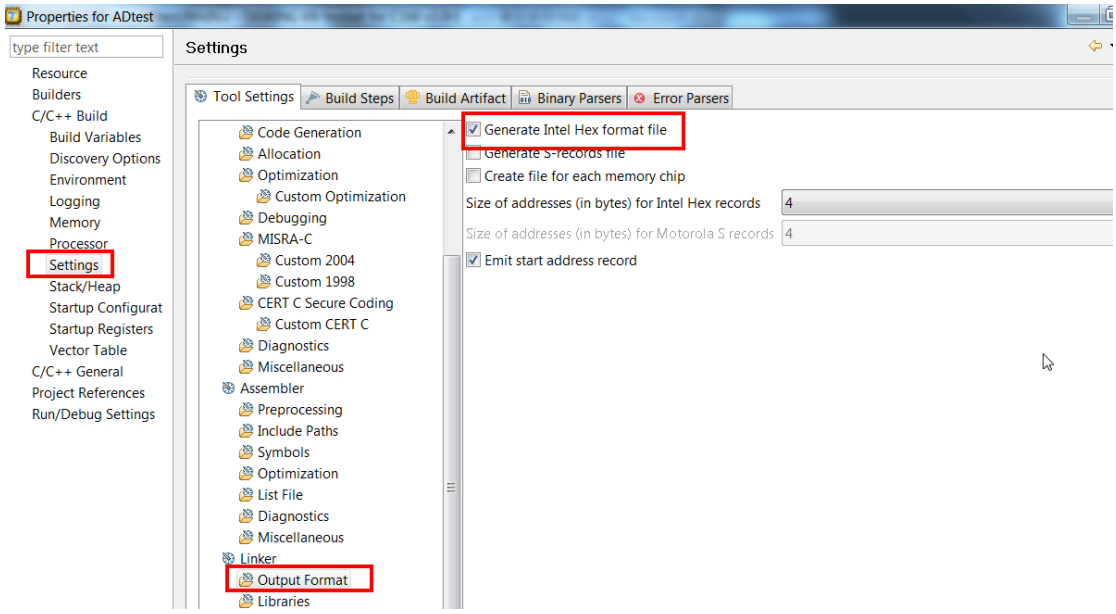


关掉刚才的 DAVE。直接从下图双击打开 DAVE，这样有修改可以直接覆盖，原来的 DAVE 就关掉吧。切记：DAVE 的生成文件是与自身.DAV 文件在同一目录下



选择项目属性





Build 以后在文件夹 debug 目录下生成 HEX

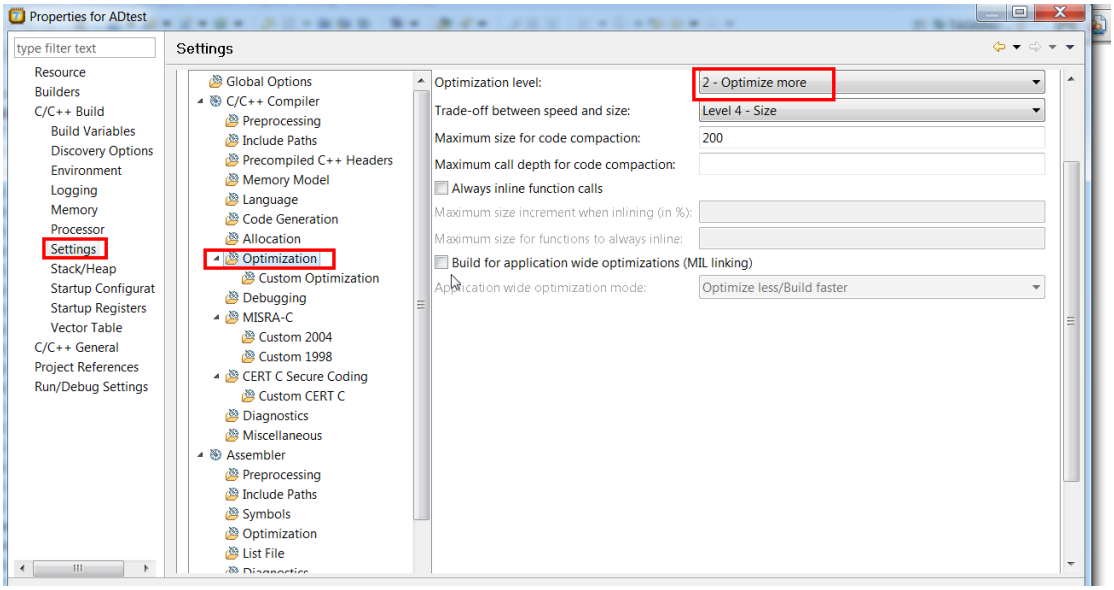
6.2 添加用户代码（XE162FL与XE160U添加代码基本相同）

在main 函数中添加下列代码（在main 函数的末尾处）

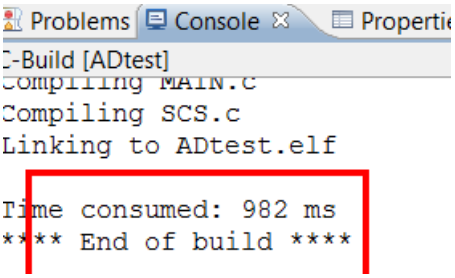
```
// USER CODE BEGIN (Main,2)
uword i,j,uwADCResult;
// USER CODE END
MAIN_vInit();

while(1)
{
// USER CODE BEGIN (Main,4)
ADC0_vStartSeq0ReqChNum(0, 0, 0,0);
while(ADC0_uwBusy());
uwADCResult=ADC0_uwGetResultData(RESULT_REG_0);
IO_vTogglePin(IO_P10_0);
for(i=0;i<uwADCResult;i++)
for(j=0;j<10000;j++);
// USER CODE END
}
```

这个演示程序里，注意关闭优化，默认是 2 级，不关闭的话，空的延时会被优化掉

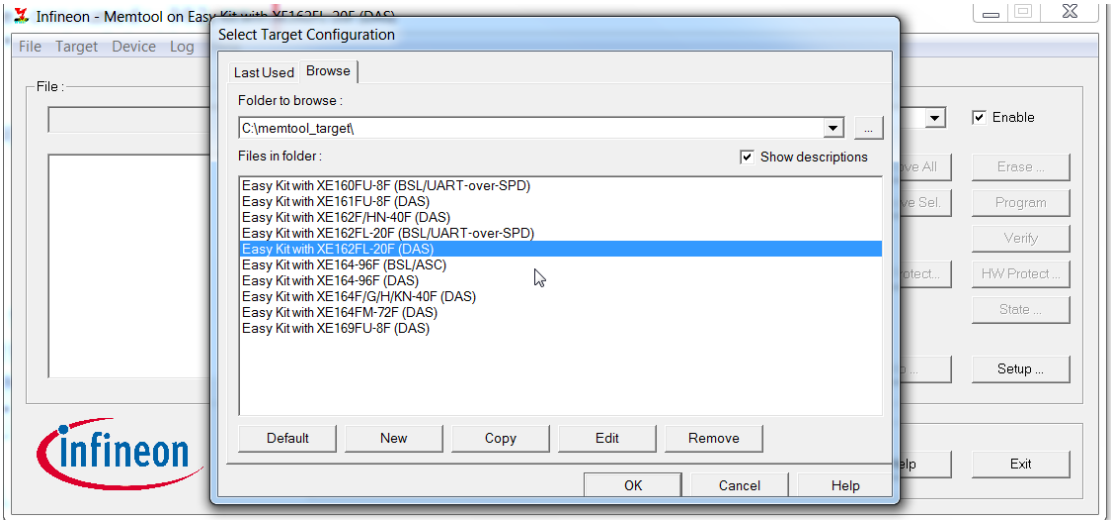


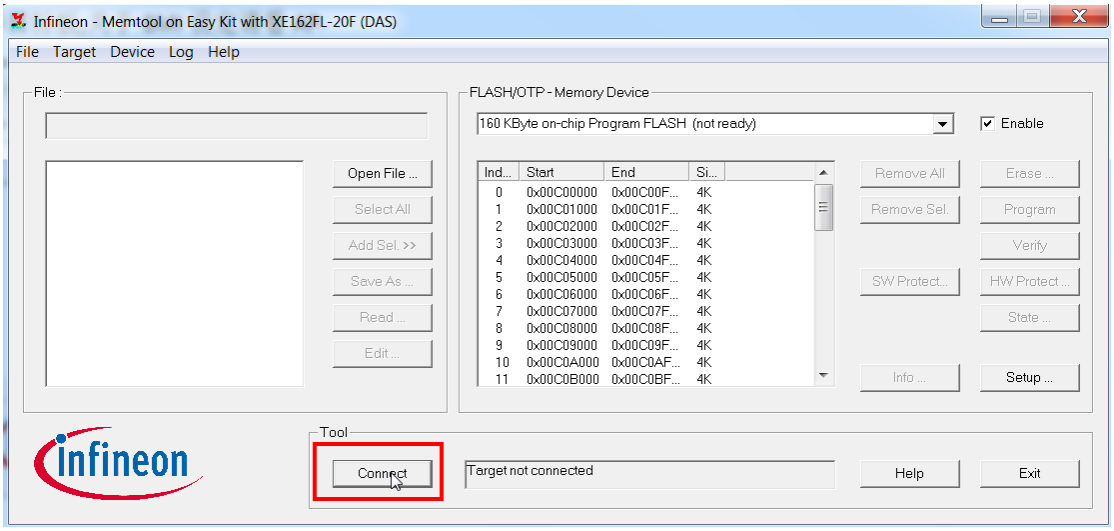
7　BUILD 以后



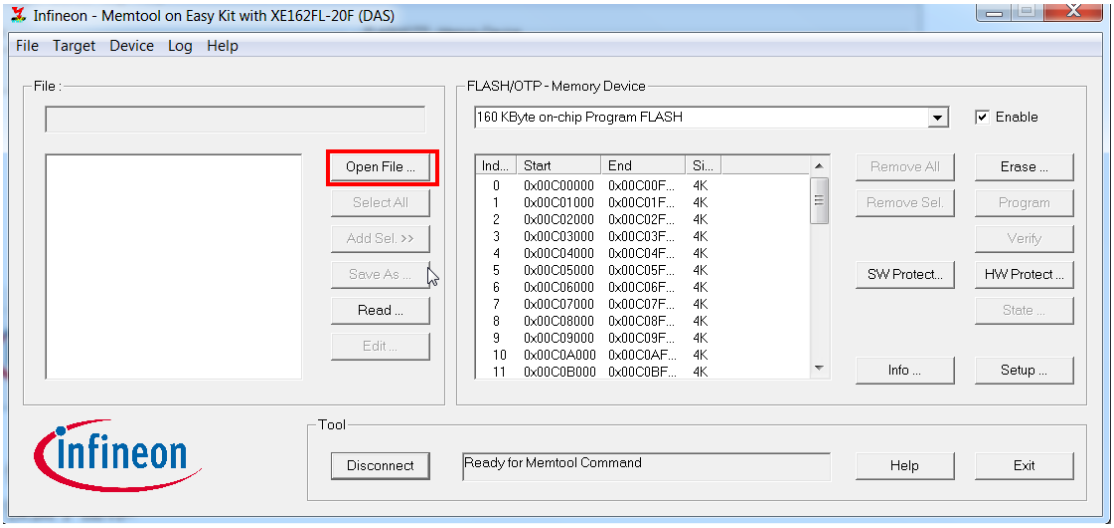
7.1 可以用 memtool 下载 HEX 到 minikit，仿真见 7.2

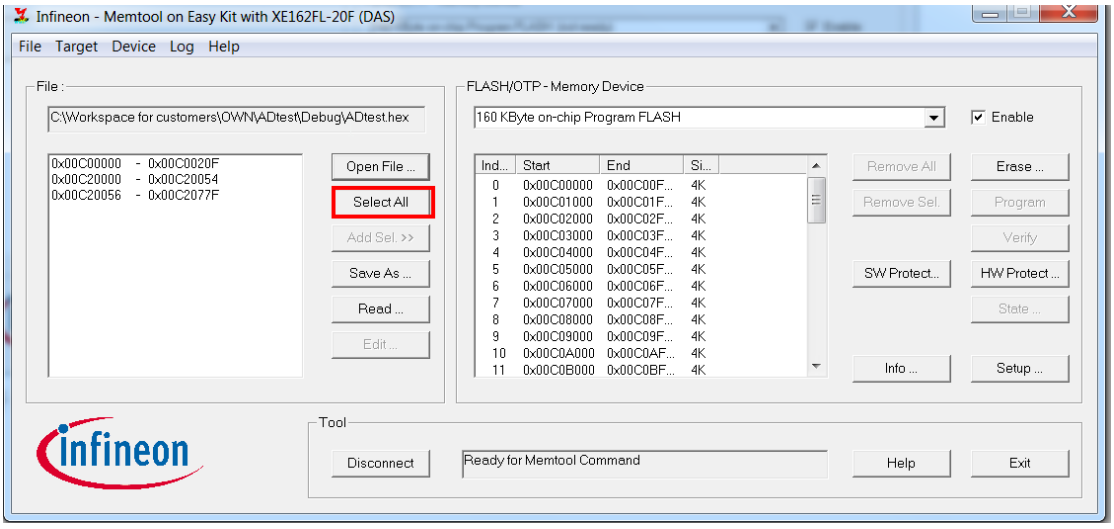
选择目标板,新片子默认 JTAG，关于 BMI 设定请看 MINIKIT 光盘 目录 说明书

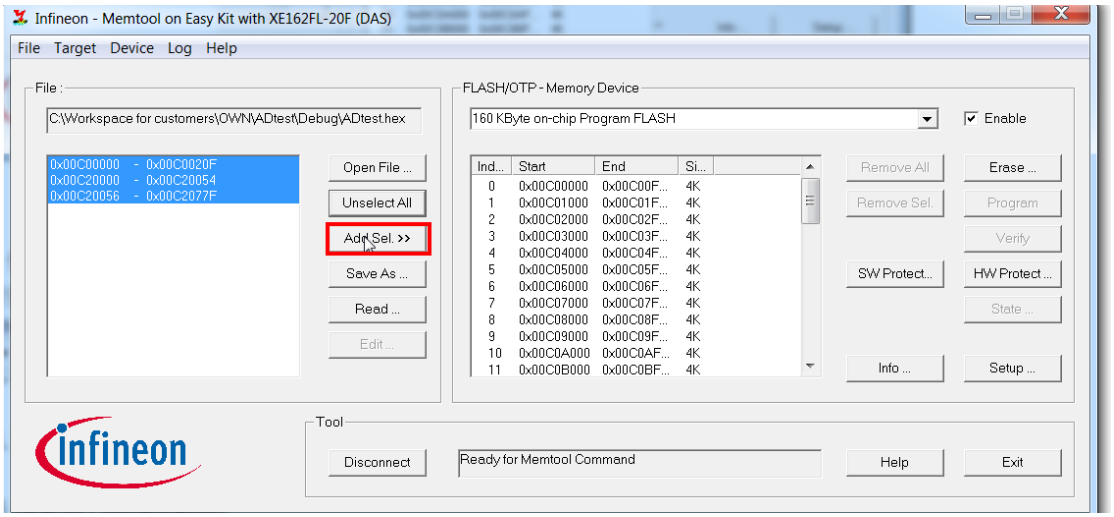


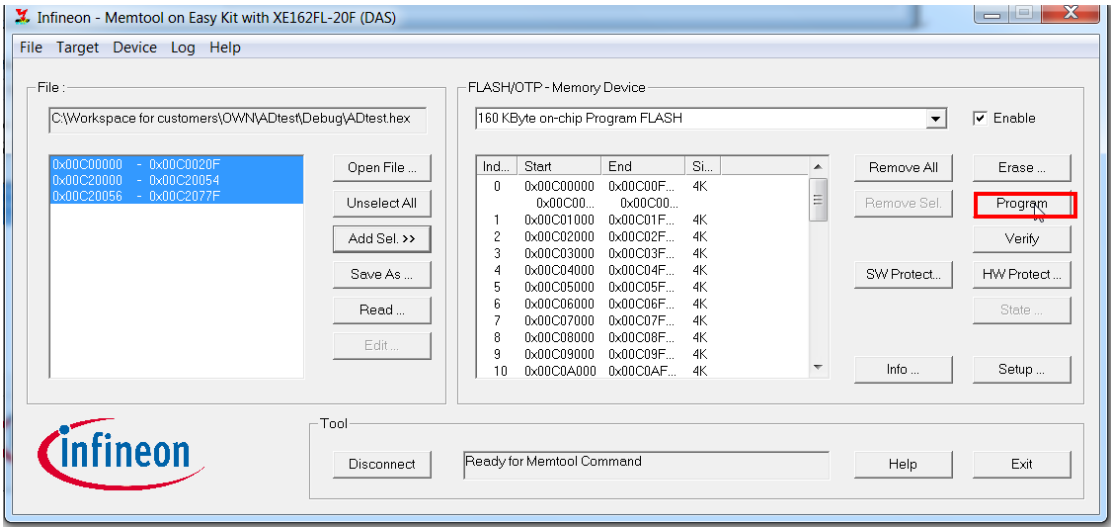

Infineon - Memtool on Easy Kit with XE162FL-20F (DAS)
File Target Device Log Help
File :
Open File ...
Select All
Add Sel. >>
Save As ...
Read ...
Edit ...
FLASH/OTP - Memory Device
160 KByte on-chip Program FLASH (not ready)
Enable
Ind... Start End Si...
0 0x00C00000 0x00C00F... 4K
1 0x00C01000 0x00C01F... 4K
2 0x00C02000 0x00C02F... 4K
3 0x00C03000 0x00C03F... 4K
4 0x00C04000 0x00C04F... 4K
5 0x00C05000 0x00C05F... 4K
6 0x00C06000 0x00C06F... 4K
7 0x00C07000 0x00C07F... 4K
8 0x00C08000 0x00C08F... 4K
9 0x00C09000 0x00C09F... 4K
10 0x00C0A000 0x00C0AF... 4K
11 0x00C0B000 0x00C0BF... 4K
Remove All
Remove Sel.
SW Protect...
Info ...
Erase ...
Program
Verify
HW Protect ...
State ...
Setup ...
Tool
Connect
Target not connected
Help
Exit

打开 HEX 文件

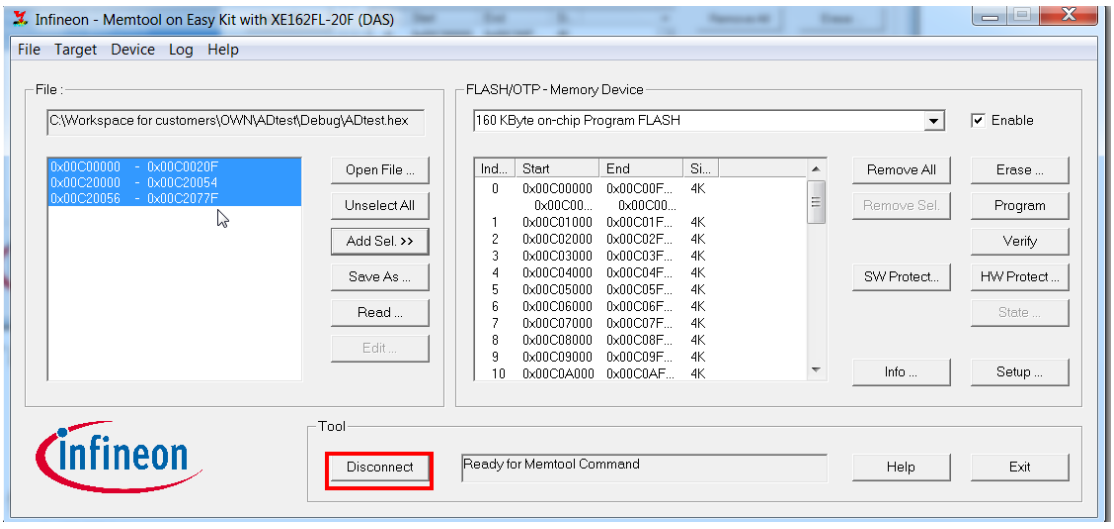

Infineon - Memtool on Easy Kit with XE162FL-20F (DAS)
File Target Device Log Help
File :
Open File ...
Select All
Add Sel. >>
Save As ...
Read ...
Edit ...
FLASH/OTP - Memory Device
160 KByte on-chip Program FLASH
Enable
Ind... Start End Si...
0 0x00C00000 0x00C00F... 4K
1 0x00C01000 0x00C01F... 4K
2 0x00C02000 0x00C02F... 4K
3 0x00C03000 0x00C03F... 4K
4 0x00C04000 0x00C04F... 4K
5 0x00C05000 0x00C05F... 4K
6 0x00C06000 0x00C06F... 4K
7 0x00C07000 0x00C07F... 4K
8 0x00C08000 0x00C08F... 4K
9 0x00C09000 0x00C09F... 4K
10 0x00C0A000 0x00C0AF... 4K
11 0x00C0B000 0x00C0BF... 4K
Remove All
Remove Sel.
SW Protect...
Info ...
Erase ...
Program
Verify
HW Protect ...
State ...
Setup ...
Tool
Disconnect
Ready for Memtool Command
Help
Exit


Infineon - Memtool on Easy Kit with XE162FL-20F (DAS)
File Target Device Log Help
File :
C:\Workspace for customers\OWN\ADtest\Debug\ADtest.hex
0x00C00000 - 0x00C0020F
0x00C20000 - 0x00C20054
0x00C20056 - 0x00C2077F
Open File ...
Select All
Add Sel. >>
Save As ...
Read ...
Edit ...
FLASH/OTP - Memory Device
160 KByte on-chip Program FLASH
Enable
Ind... Start End Si...
0 0x00C00000 0x00C00F... 4K
1 0x00C01000 0x00C01F... 4K
2 0x00C02000 0x00C02F... 4K
3 0x00C03000 0x00C03F... 4K
4 0x00C04000 0x00C04F... 4K
5 0x00C05000 0x00C05F... 4K
6 0x00C06000 0x00C06F... 4K
7 0x00C07000 0x00C07F... 4K
8 0x00C08000 0x00C08F... 4K
9 0x00C09000 0x00C09F... 4K
10 0x00C0A000 0x00C0AF... 4K
11 0x00C0B000 0x00C0BF... 4K
Remove All
Remove Sel.
SW Protect...
Info ...
Erase ...
Program
Verify
HW Protect ...
State ...
Setup ...
Tool
Disconnect
Ready for Memtool Command
Help
Exit





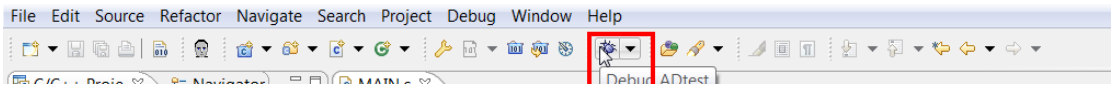
完了以后断开连接



按下复位键就可以了。

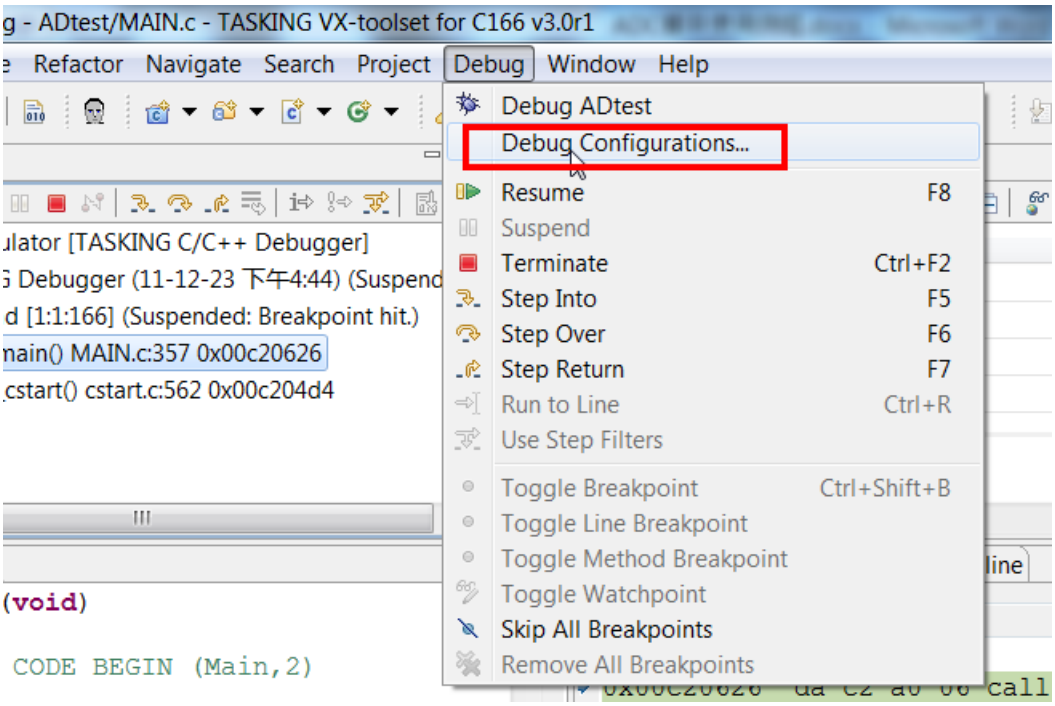
7.2 仿真

或者按虫子图标

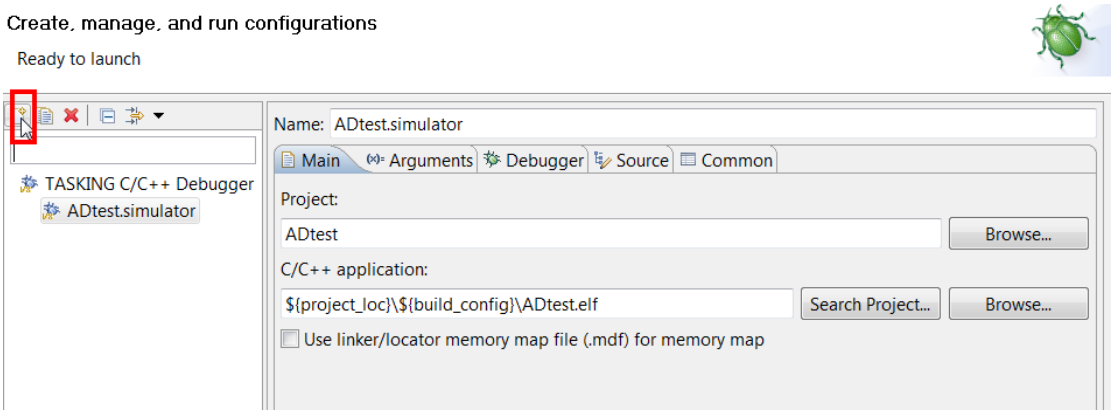


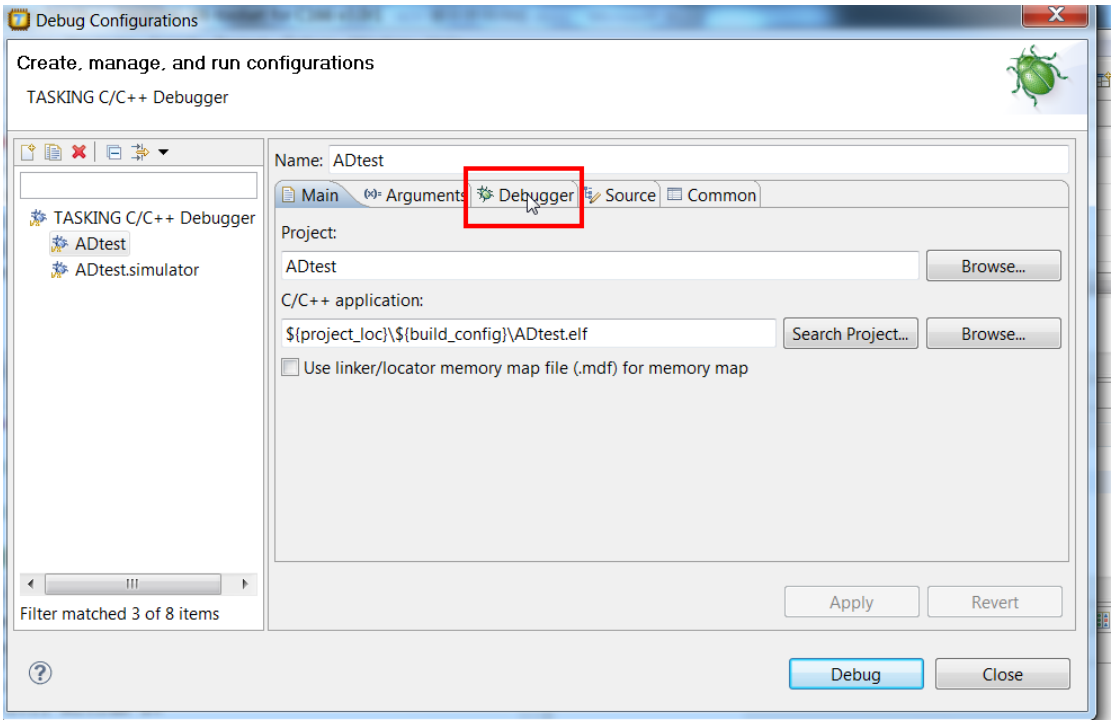
默认 Simulator

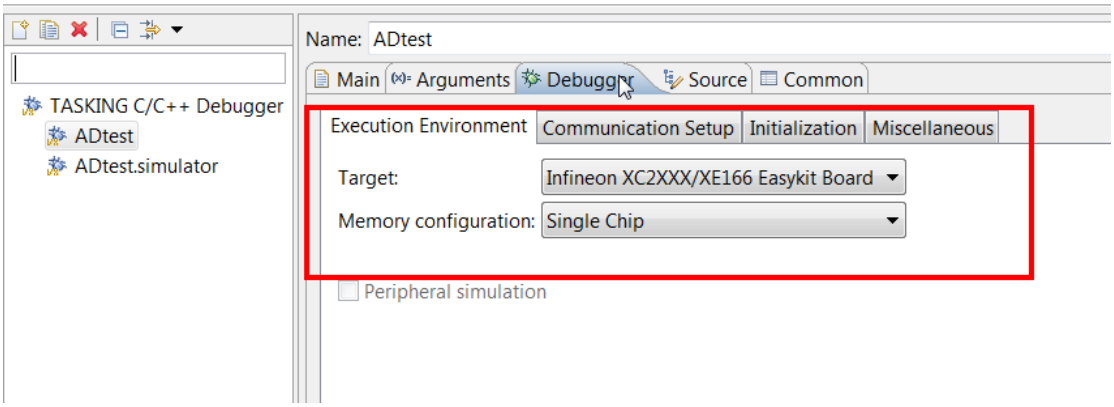
选择 DEBUG Configuration

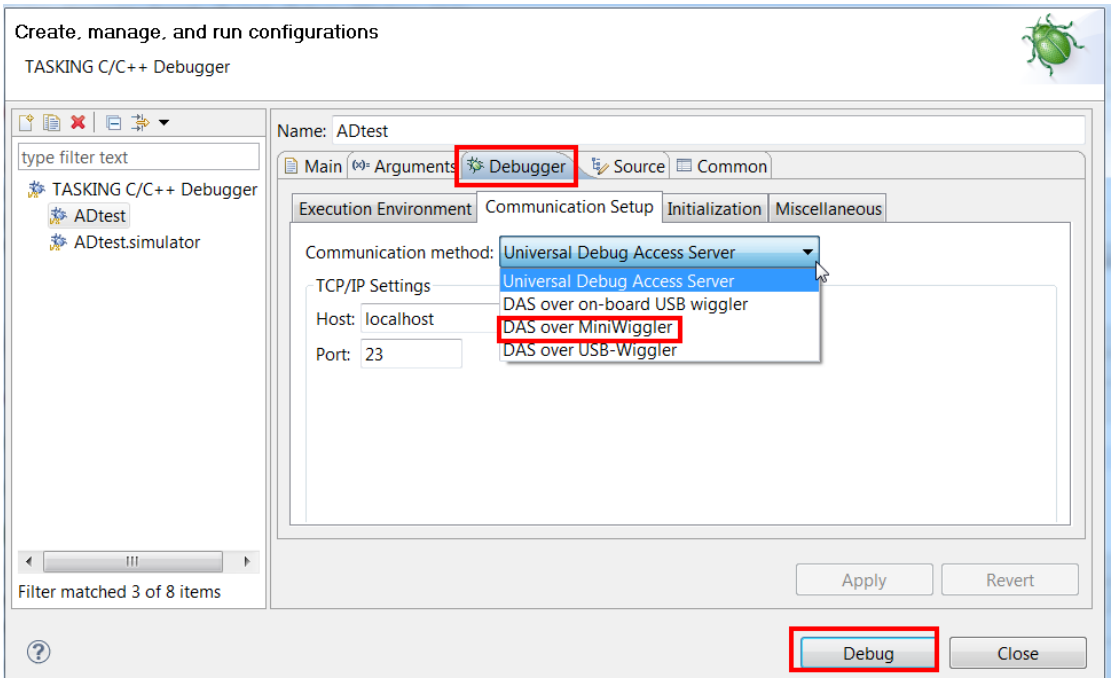


点击新建，设定一个新的仿真设定




Debug Configurations
Create, manage, and run configurations
TASKING C/C++ Debugger
TASKING C/C++ Debugger
ADtest
ADtest.simulator
Filter matched 3 of 8 items
Name: ADtest
Main
Arguments
Debugger
Source
Common
Project:
ADtest
Browse...
C/C++ application:
${project_loc}\${build_config}\ADtest.elf
Search Project...
Browse...
Use linker/locator memory map file (.mdf) for memory map
Apply
Revert
Debug
Close


TASKING C/C++ Debugger
ADtest
ADtest.simulator
Name: ADtest
Main
Arguments
Debugger
Source
Common
Execution Environment
Communication Setup
Initialization
Miscellaneous
Target:
Infineon XC2XXX/XE166 Easykit Board
Memory configuration:
Single Chip
Peripheral simulation


Create, manage, and run configurations
TASKING C/C++ Debugger
type filter text
TASKING C/C++ Debugger
ADtest
ADtest.simulator
Filter matched 3 of 8 items
Name: ADtest
Main
Arguments
Debugger
Source
Common
Execution Environment
Communication Setup
Initialization
Miscellaneous
Communication method:
Universal Debug Access Server
TCP/IP Settings
Host: localhost
Port: 23
Universal Debug Access Server
DAS over on-board USB wiggler
DAS over MiniWiggler
DAS over USB-Wiggler
Apply
Revert
Debug
Close

观察变量，注意是全局变量